Rancang Bangun Aplikasi Pelacak Bibit Kelapa Sawit Berbasis Aplikasi *Mobile*

Lukmannil Hakim[1], Feri Candra[2]

[1]Program Studi Teknik Informatika, Universitas Riau,Pekanbaru, Indonesia
[2]Jurusan Teknik Elektro, Universitas Riau, Pekanbaru, Indonesia

E-mail: lukmannil_hakim5719@unri.ac.id

Abstract

The success of oil palm cultivation is deeply depend by quality of the seedlings, and for this reason the implementation of nurseries needs a special attention. The problem that occurs regularly is that the amount of seeds sent from the nursery area is not equal to the amount of seeds contained in the farm. The number of seeds that are not allocated causes misfortunes in the organization. Seedlings are also defenseless and can be replaced by different seeds on their way. An application is required and you can work with a lot of data. This application is expected to have the option to record status of the palm oil seedling area productively. Ionic is a system that can create hybrid mobile applications. This versatile application is built with commonly used programming language for the web (HTML, CSS, and JavaScript). This application will be used in places with network problems. Documenting changes in seed information status provides types of seed information assistance to the Board. This application works with both online and offline databases.

Keywords: Palm oil, nursery, mobile hybrid application, data management

1. Pendahuluan

PT Perkebunan Nusantara (PTPN) V adalah sebuah perusahaan yang bergerak di bidang perkebunan di Provinsi Riau. Keberhasilan penanaman kelapa sawit sangat bergantung pada kualitas bibit, sehingga teknis pelaksanaan persemaian memerlukan perhatian khusus[1]. Permasalahan yang sering terjadi adalah jumlah benih yang dikirim dari lokasi persemaian tidak sesuai dengan jumlah benih yang diterima di kebun. Berkurangnya jumlah benih selama pengirimanakan merugikan PTPN V. Penelitian yang dilakukan penulis berasal dari sistem pelacakan barang. Pengguna dapat melacak lokasi barang dengan memasukkan nomor resi di sistem. Kemudian, sistem akan memeriksa status posisi terakhir produk di *database* menggunakan nomor resi yang dimasukkan oleh pengguna. Status lokasi benih terakhir akan selalu berubah, dan setiap *item* berhenti di setiap pos pemeriksaan[2]. Nomor ID pada benih digunakan sama dengan nomor resi. Namun, sistem dapat menggunakan kode *Quick Response* (QR) untuk memudahkan pengguna untuk masuk.Kode QR adalah gambar dua dimensi yang dapat digunakan untuk menyimpan data teks, mulai dari angka, alfanumerik, dan kode biner. Kode QR pertama kali diperkenalkan pada tahun 1994 oleh Denso Corporation, sebuah perusahaan Jepang yang bergerak di bidang industri otomotif. Kode QR juga dikenal sebagai simbol yang menandai objek berupa batang hitam[3]. Kode QR dapat digunakan untuk mengidentifikasi benih kelapa sawit dan digunakan hingga ditanam.

2. Metodologi Penelitian

2.1. Metode Pengembangan Sistem

Penulis menggunakan metode *waterfall* dalam penelitian ini. Metode *waterfall* merupakan model yang sistematis dan linier, artinya sistem tersebut dibangun secara berurutan[4].

2.2. Analisis Kebutuhan Sistem

Penulis melakukan beberapa tahap yaitu analisis kebutuhan *input*, analisis kebutuhan *output*, analisis kebutuhan proses, analisis kebutuhan perangkat lunak, dan analisis kebutuhan perangkat keras.

2.2.1. Analisis Kebutuhan *Input*

Pengguna sistem adalah pegawai PTPN V yang bertanggung jawab merawat, mengirim dan menerima bibit. Pekerja harus memiliki nama pengguna dan kata sandi untuk masuk. Ketika pengguna memindai kode QR benih, data benih dapat dilihat. Ketika sulit untuk mengakses internet, pengguna dapat memasukkan data *seed* ke dalam *database online* (Firebase) atau *database*

offline (SQLite). Firebase adalah DBMS unik, berbeda dari *database* lain. Setiap *database* Firebase disimpan dalam bentuk Javascript Object Notation (JSON)[5]. SQLite adalah *database open source* yang dirancang khusus untuk menyimpan data pada perangkat dengan memori terbatas[6].

2.2.2. Analisis Kebutuhan *Output*

Kebutuhan *output* dari sistem adalah kemampuan untuk menampilkan data setelah memindai kode QR benih melalui sistem. Ini juga membutuhkan sistem untuk dapat menampilkan pemberitahuan setelah menambahkan, mengubah atau menghapus data dalam *database*.Selain itu, karyawan PTPN V harus menampilkan sistem dengan mudah dan dimengerti.

2.2.3. Analisis Kebutuhan Proses

Proses yang dibutuhkan untuk sistem adalah sebagai berikut:

1. Proses *login*.
2. Proses *loading pages*.
3. Proses pemindaian kode QR.
4. Proses *action loading*.

2.2.4. Analisis Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak dalam sistem ini terdiri dari dua sisi, yaitu sisi *developer* dan sisi pengguna. Kebutuhan perangkat lunak pada sisi *developer* sebagai berikut:

1. *Text editor*.
2. *Browser*.
3. *Smartphone*.
4. *Command Prompt*.
5. Node.js. Node.js adalah perangkat lunak yang dirancang untuk mengembangkan aplikasi berbasis web menggunakan JavaScript[7].
6. Ionic dan Apache Cordova. Ionic adalah *framework* yang dapat menggunakan teknologi web (seperti HTML5, CSS dan JavaScript) untuk membuat aplikasi *mobile*. Apache Cordova adalah teknologi yang dapat membuat aplikasi web terlihat seperti aplikasi *native*[8].

Kebutuhan perangkat lunak minimum untuk pengguna adalah sistem operasi Android 4.4 atau iOS 8 pada *smartphone*. Kedua sistem operasi ini dipilih karena kebanyakan orang menggunakan sistem operasi ini[9].

2.2.5. Analisis Kebutuhan Perangkat Keras

Kebutuhan perangkat keras dalam sistem ini terdiri dari komputer dan *smartphone*. Komputer digunakan dalam proses *system developing*. Spesifikasi *smartphone* yang digunakan penulis untuk penelitian ini yaitu:

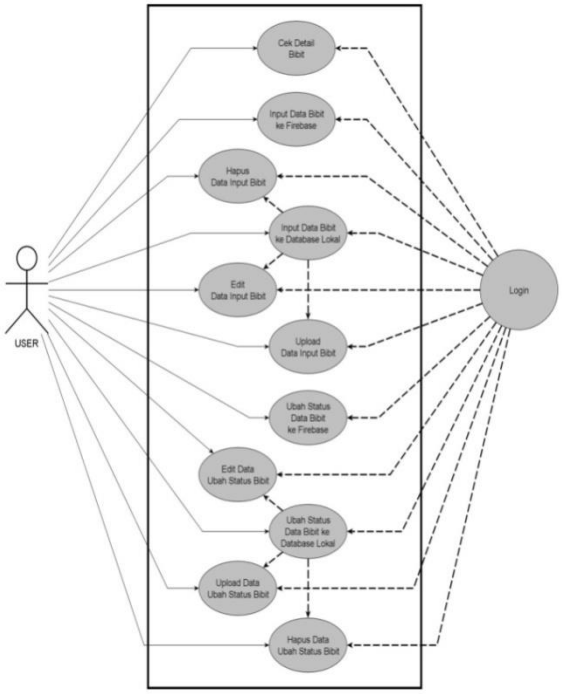
1. Android *version* 8.0.0
2. Penyimpanan internal 32 *Gigabyte*
3. *Random Access Memory* 4 *Gigabyte*
4. Kamera depan dengan resolusi 19 *Megapixel*

2.3. Perancangan

Perancangan sistem meliputi dua bagian yaitu *use case diagram* dan *flow chart design*. Metode yang digunakan dalam perancangan adalah metode perancangan data terarah karena mudah dipahami dan efisien. Metode ini menjelaskan keseluruhan sistem yang akan dibuat.

2.3.1. *Use Case Diagram*

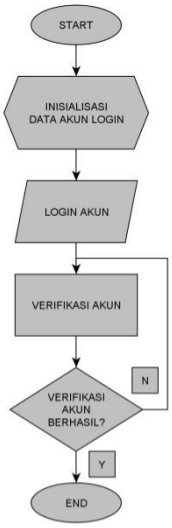
Perancangan *Use Case Diagram* digunakan penulis untuk mendeskripsikan interaksi antara pengguna dengan sistem, dapat dilihat pada Gambar 1.



Gambar 1. Rancangan sistem dijabarkan dengan *Use Case Diagram*

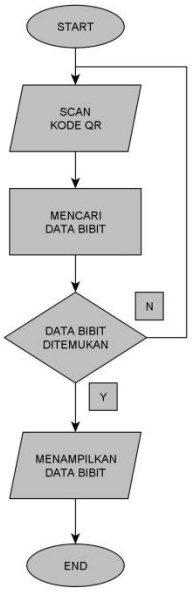
2.4.2. *Flowchart*

Perancangan sistem dengan *flowchart* menggambarkan dan menyederhanakan rangkaian proses, sehingga urutan berdasarkan proses tersebut mudah dipahami. Gambar 2 menunjukkan *flowchart* halaman *login*. Setelah verifikasi akun berhasil, pengguna akan diarahkan ke beranda.



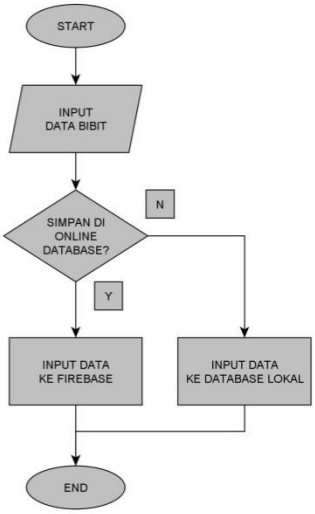
Gambar 2. Flowchart pada halaman *login*

Gambar 3 menunjukkan diagram alir dari fungsi pemeriksaan status benih. Pengguna memindai kode QR yang berisi kode benih. Setelah dilakukan pemindaian, sistem akan mencari kode *seed* di *database*. Jika kode *seed* ditemukan, pengguna akan diarahkan ke halaman data *seed*.



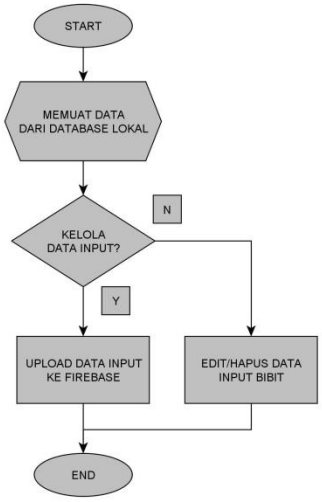
Gambar 3. *Flowchart* proses *scan* dan pemindaian data bibit

Gambar 4 menunjukkan diagram alir halaman *input* benih. Pengguna dapat memilih untuk menyimpan data *input* langsung ke *database online* (Firebase) atau *databaseoffline* (SQLite).



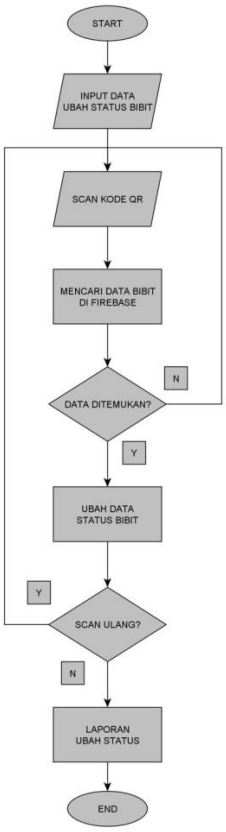
Gambar 4. *Flowchart* pada halaman penambahan data bibit

Gambar 5 menunjukkan diagram alir halaman penyimpanan *offline* untuk data *input seed*. Pengguna dapat mengedit, menghapus dan mengunggah data masukan benih.



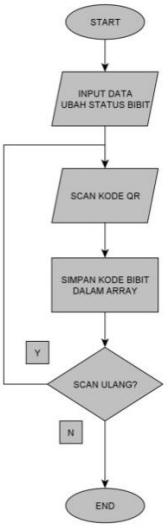
Gambar 5. *Flowchart* pada halaman penyimpanan data *input* bibit di *offlinedatabase*

Gambar 6 menunjukkan diagram alir untuk mengubah status benih ke halaman *database online*. Pengguna memasukkan data status perubahan benih sebelum mulai memindai kode QR benih. Setelah memindai kode QR, sistem akan mencari data *seed*. Jika ditemukan data benih maka status benih akan berubah.



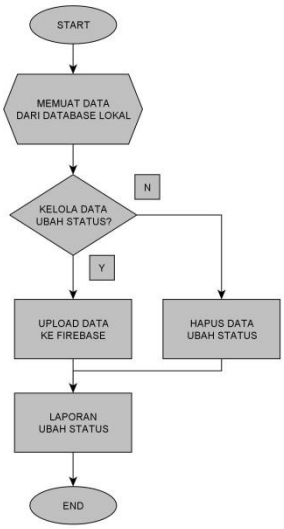
Gambar 6. *Flowchart* pada halaman pengubahan status bibit menuju *online database*

Gambar 7 menunjukkan diagram alir halaman yang mengubah status seed ke *database offline*. Pengguna memasukkan data status perubahan benih sebelum mulai memindai kode QR benih. Tidak seperti proses mengubah status *seed* ke *database online*, setelah pemindaian, sistem tidak memeriksa kode *seed* dan langsung menyimpannya dalam *array*. Periksa kode benih pada halaman penyimpanan data *offline* untuk mengubah status benih.



Gambar 7. *Flowchart* pada halaman pengubahan status bibit menuju *online database*

Gambar 8 menunjukkan diagram alir untuk mengubah status benih dari halaman penyimpanan data *offline*. Pengguna memasukkan data sebelum mulai memindai kode QR untuk mengubah status benih. Pengguna hanya dapat memilih untuk mengunggah atau menghapus data perubahan status benih.



Gambar 8. *Flowchart* pada halaman penyimpanan *offline* data pengubahan status bibit

2.4.3. Perancangan Database

Basis data dirancang struktur yang baik, sehingga dapat digunakan dalam waktu yang lama.Penulis menggunakan Firebase dan SQLite.Objek pada Tabel 1 berisi data dari setiap benih yang terdaftar di sistem. Saat memeriksa data *seed* di aplikasi, data objek akan ditampilkan.

Tabel 1. Objek master.json yang memuat data primer bibit

Key	*Type*	*Value*
id	*string*	identitas bibit
idgroup	*string*	identitas grup bibit saat dimasukkan ke *database*
asal	*string*	kode asal bibit
klmpk	*string*	kode kelompok pembibitan
lokasi	*string*	kode lokasi bibit terkini
serbuk	*number*	nomor penyerbukan
tgl	*string*	tanggal data bibit dimasukkan ke *database*
varietas	*string*	kode varietas bibit

Objek pada tabel 2 memuat dan mendokumentasikan perubahan data pada objek master.json. Aplikasi tidak dapat mengubah dan menghapus data pada objek ini.

Tabel 2. Objek status.json yang memuat perubahan data status bibit

Key	*Type*	*Value*
id	*string*	identitas perubahan status bibit
id_bibit	*string*	identitas bibit
ket	*string*	keterangan dalam perubahan status bibit
lokasi_awal	*string*	kode lokasi awal bibit sebelum diubah
lokasi_akhir	*string*	kode lokasi bibit setelah diubah
stat	*string*	kode status bibit yang berisi keterangan afkir (mati), dalam pengiriman, dalam pembibitan, atau dalam kebun (sudah ditanam)
waktu	*string*	tanggal dan waktu perubahan status bibit

Objek pada tabel 3 memuat data pengiriman bibit.Data ini digunakan dalam laporan pengiriman bibit.

Tabel 3. Objek kirbit.json yang memuat data pengiriman bibit

Key	*Type*	*Value*
id	*string*	identitas pengiriman bibit
tujuan	*string*	kode lokasi tujuan pengiriman bibit
waktu	*string*	tanggal dan waktu pengiriman bibit
nm_supir	*string*	nama sopir pengiriman bibit
nm_kawal	*string*	nama pengawal pengiriman bibit
no_ken	*string*	nomor kendaraan pengiriman
no_spb	*string*	nomor surat permintaan barang
r_afdel	*string*	kode afdeling bibit
r_blok	*string*	kode blok bibit

Objek pada Tabel 4 berisi data tentang tempat menempatkan benih. Data tersebut dapat mencakup pembibitan dan lokasi penanaman. Lokasi persemaian terbagi menjadi dua yaitu *Pre Nursery* (PN) dan *Main Nursery* (MN).

Tabel 4. Objek lokasi.json yang memuat data lokasi pembibitan

Key	*Type*	*Value*
Id	*string*	identitas lokasi bibit
Jenis	*string*	kode lokasi pembibitan, data dapat berisi pn, mn, atau *null*
Kebun	*string*	nama lokasi bibit
Kode	*string*	kode lokasi bibit yang lebih spesifik
Ket	*string*	keterangan dan deskripsi lokasi bibit

Sistem menggunakan SQLite sebagai *database offline*.Tabel 5 berisi data tentang perubahan status benih. Kode benih disimpan dalam *array* untuk menyimpan banyak benih dalam variabel.

Tabel 5. Tabel ubah status untuk menyimpan data status bibit

Name	*Type*	*Value*
Id	*integer*	nomor identitas ubah status bibit (*primary key*, *auto increment*)
id_bibit	*text*	identitas bibit, berbentuk *array* atau kompilasi bibit-bibit yang dipindai
Ket	*text*	keterangan perubahan status bibit
lokasi_awal	*text*	kode lokasi awal bibit sebelum diubah
lokasi_akhir	*text*	kode lokasi bibit setelah diubah
stat	*text*	kode status bibit yang berisi keterangan afkir (mati), dalam pengiriman, dalam pembibitan, atau dalam kebun (sudah ditanam)
waktu	*text*	keterangan waktu perubahan status

Tabel 6 berisi data masukan benih. Pengguna akan menyimpan data *seed* dan kuantitas yang akan dimasukkan ke dalam Firebase.

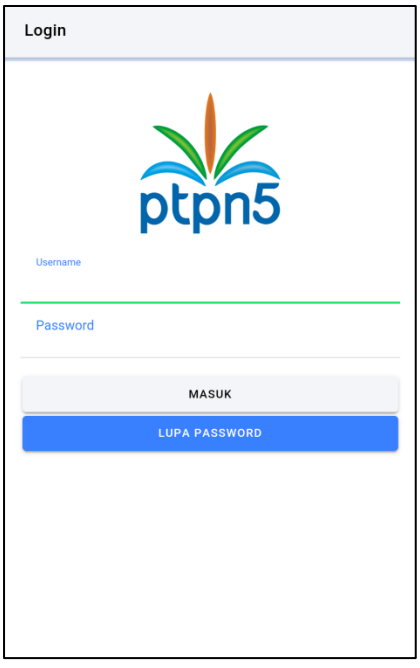
Tabel 6. Tabel tambah bibit untuk menyimpan data jumlah bibit

Key	*Type*	*Value*
jumlah	*integer*	jumlah bibit yang ditambahkan
idgroup	*text*	identitas grup bibit saat dimasukkan ke *database* (*primary key*)
asal	*text*	kode asal bibit
klmpk	*text*	kode kelompok pembibitan
lokasi	*text*	kode lokasi bibit terkini
serbuk	*text*	nomor penyerbukan
tgl	*text*	tanggal data bibit dimasukkan
varietas	*text*	kode varietas bibit

3. Hasil dan Pembahasan

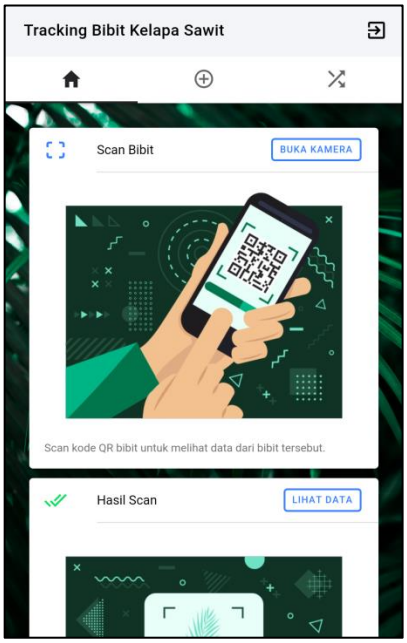
3.1. *User Interface*

UI adalah tampilan grafik dari sebuah komputer, dan operasi serta metode pemrogramannya membuatnya dapat dimengerti oleh pengguna[10]. Penulis membuat beberapa halaman untuk memisahkan berbagai fungsi kompleks dalam sistem. Halaman *login* adalah halaman pertama yang dapat dikunjungi pengguna.Pengguna dapat masuk secara *online* atau *offline*.UI halaman *login* dapat dilihat pada Gambar 9.



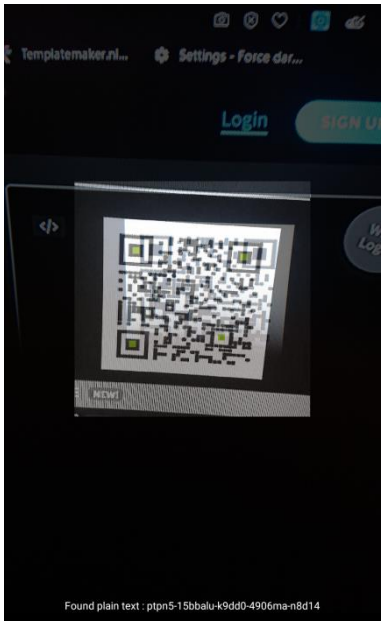
Gambar 9. Tampilan sistem memuat halaman *login* sistem

Setelah berhasil *login*, pengguna akan diarahkan ke halaman utama. Halaman ini berisi 3 *tab* yaitu *tab* utama, *tab* tambah data dan *tab* ubah status data. UI halaman menu utama dapat dilihat pada Gambar 10, Gambar 11 dan Gambar 12.

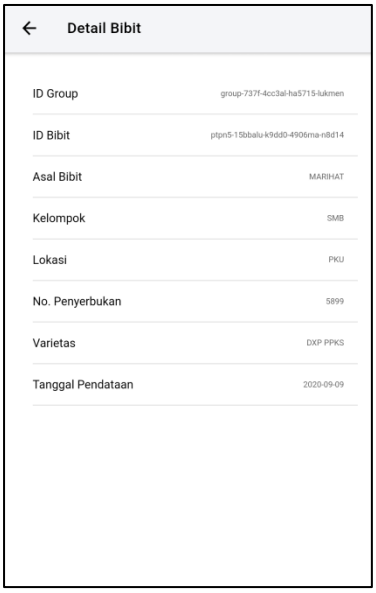


Gambar 10. Tampilan menu utama sistem pada *tab* utama setelah *login*

Pada halaman menu utama, terdapat fungsi pemindaian benih.Fitur ini digunakan untuk memindai kode QR benih dan kemudian mengarahkan pengguna ke halaman yang dipindai untuk melihat data benih secara rinci. UI halaman yang dipindai dapat dilihat pada Gambar 11 dan Gambar 12.

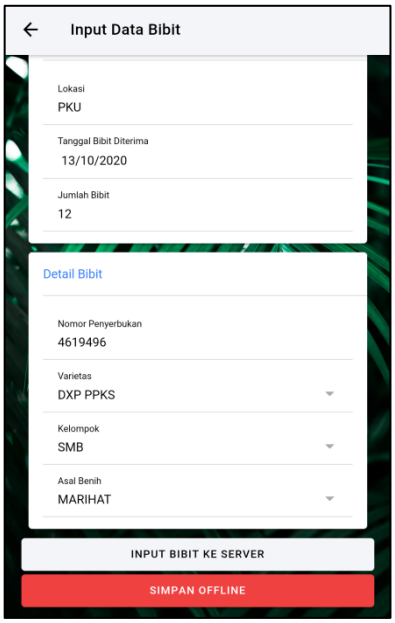


Gambar 11. Tampilan sistem saat melakukan pemindaian



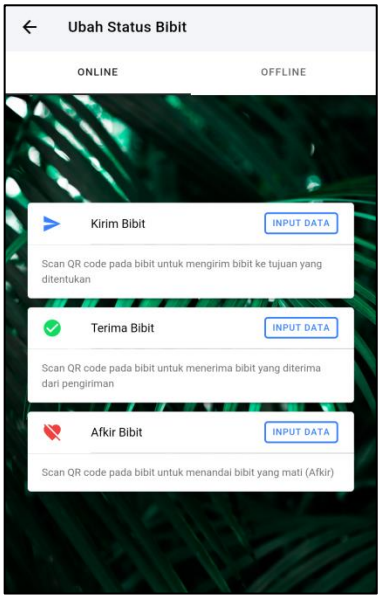
Gambar 12. Tampilan sistem detail data setelah pemindaian

Halaman ini digunakan untuk menambah jumlah data benih.Pengguna dapat langsung menambahkan data ke *database online*, atau menyimpan data yang ditambahkan ke *database offline* terlebih dahulu. UI halaman penambahan data dapat dilihat pada Gambar 13.



Gambar 13. Tampilan sistem pada halaman penambahan data

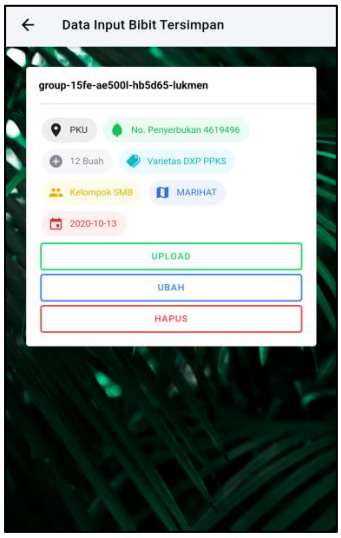
Halaman *status seed change* memiliki dua *tab*, yaitu *tab online* dan *tab offline*, sebagai acuan untuk keperluan penyimpanan data. Setiap *tab* berisi perubahan status pengiriman, penerimaan dan penolakan benih. UI halaman konversi data dapat dilihat pada Gambar 16 dan Gambar 14.



Gambar 14. Tampilan sistem pada halaman pengubahan status data

Setelah pengguna menyimpan data ke *database offline*, data tersebut dapat dilihat di halaman penyimpanan *offline*.Halaman penyimpanan *offline* untuk menambah dan mengedit data ditampilkan secara terpisah.UI

halaman penyimpanan *offline* dapat dilihat pada Gambar 15.



Gambar 15. Tampilan sistem pada halaman pengubahan status data *tab offline*

3.2. Pengujian

Penulis menggunakan pengujian *black box* untuk menguji sistem oleh orang-orang yang hanya mengetahui fungsi dari setiap komponen fungsional sistem tetapi tidak mengetahui berbagai proses kinerja dari setiap komponen fungsional. Pengujian dilakukan dengan memberikan kuesioner kepada 6 karyawan PTPN V. Hasil kuisioner akan menunjukkan kebenaran sistem. Hasil pengujian sistem ditunjukkan pada Tabel 7.

Tabel 7. Tabel hasil *blackbox testing*

No	Pernyataan	Berhasil	Tidak Berhasil
1	Pengguna dapat melakukan *login* dengan akun masing-masing	6	
2	Pengguna dapat menambahkan data bibit ke *online database*	6	
3	Pengguna dapat menambahkan data bibit ke *offline database*	6	
4	Pengguna dapat mengunggah data dari *offline database* ke *online database*	6	
5	Pengguna dapat melihat data bibit dengan menggunakan fitur QR *code scan*	6	
6	Pengguna dapat mengubah status bibit dan mengirim perubahan data ke *online database*	6	
7	Pengguna dapat mengubah status bibit dan mengirim perubahan data ke *offline database*	6	
8	Sistem dapat mengatasi kesalahan dalam penambahan data bibit	6	
9	Sistem dapat mengatasi kesalahan dalam pemindaian QR code bibit	6	

Hasil pengujian *black box* sistem ini menunjukkan bahwa semua fungsi sistem beroperasi secara normal.

4. Kesimpulan

Berdasarkan hasil penelitian rancang bangun aplikasi pelacak bibit kelapa sawit berbasis aplikasi *mobile*, maka dapat disimpulkan bahwa sistem yang dibangun dapat menggunakan dua jenis *database* yaitu Firebase sebagai *database online* dan SQLite sebagai *database offline*. Pengguna dapat masuk dengan akun yang disediakan oleh *administrator*. *Administrator* dapat melihat aktivitas pengguna yang mengubah data di *database*. Dokumentasi perubahan data status benih memberikan kemudahan untuk mengelola data benih. Rekomendasi dari penelitian ini adalah metode *white box* dapat digunakan untuk meningkatkan performansi sistem, sehingga meningkatkan kecepatan dan efektivitas proses scanning dan pengujian sistem.

Referensi

[1]. Rosa, Rahayu Novrina dan Sofyan Zaman, "Pengelolaan Pembibitan Tanaman Kelapa Sawit (Elais guineensis Jacq.) di Kebun Bangun Bandar, Sumatera Utara. Management Oil Palm (Elaeis guineensis Jacq.) Nursery in Bangun Bandar Estate, North Sumatera", vol. 5, pp. 325-333, Nov. 2017.

[2]. Sitanggang, Dewi Yeni, *et al*, "Aplikasi Sistem Pengecekan Tarif Dan Tracking Status Pengiriman Barang Menggunakan Ponsel Berbasis Java (Studi Kasus: PT JNE Cirebon)", vol. 4, pp. 122-135, Nov. 2014.

[3]. Nugraha, M Pasca, "Pengembangan Aplikasi QR Code Generator dan QR Code Reader dari Data Berbentuk

Image" *in Konferensi Nasional Informatika*, STEI ITB – Bandung, 2011. pp. 148-155.
[4]. Rahayu, Yuni, *et al*, "Pemetaan Penyebaran dan Prediksi Jumlah Penduduk Menggunakan Metode Geometrik di Wilayah Bandar Lampung Berbasis Web-GIS", vol. 2, pp. 95-101, May. 2016.
[5]. Cheng, Fu, Building Mobile Apps with Ionic 2 and Firebase. Auckland: Apress Media, 2017.
[6]. Junindar, Xamarin Forms: Membangun Aplikasi Mobile Cross-Platform (Android, iOS & Windows). Yogyakarta: Ebooku, 2017.
[7]. Nugroho, M. Agung dan Rikie Katardi, "Analisis Kinerja Penerapan Container untuk Load Balancing Web Server pada Raspberry Pi", vol. 1, pp. 7-15, Dec. 2016.
[8]. Yusuf, Sani, Ionic Framework By Example. Birmingham: Pack Publishing Ltd., 2016.
[9]. Irwansyah, Edy dan Jurike V. Moniaga, Pengantar Teknologi Informasi. Yogyakarta: Deepublish, 2014.
[10]. Aziza, Rifda Faticha Alfa dan Yahya Taufiq Hidayat, "Analisa Usability Testing Desain User Interface pada Website Tokopedia Menggunakan Metode Heuristics Evaluation", vol. 13, pp. 7-11, 2019.